第 11 卷 第 4 期　运 筹 与 管 理　Vol.11, No.4
2002 年 8 月　OPERATIONS RESEARCH AND MANAGEMENT SCIENCE　Aug., 2002

约束装箱问题的混合遗传算法求解

陈迎春， 吴晓平， 宋业新

（海军工程大学 基础部，湖北 武汉 430033）

摘 要：本文将最佳适应法和遗传算法相结合，提出了一种新的启发式混合遗传算法对具有时间约束的装箱问题进行求解，给出了具体的算法步骤，试算结果表明基于启发式算法的混合遗传算法适合于求解各种约束条件下的大规模装箱问题。

关键词：混合遗传算法；装箱问题；约束

中图分类号：O224　文章标识码：A　文章编号：1007-3221(2002)04-0021-05

Hybrid Genetic Algorithm for Bin-packing Problem with Constraints

CHEN Ying-chun, WU Xiao-ping, SONG Ye-xin

(*Dept. of Basic Courses, Naval Univ, of Engineering, Wuhan* 430033, *China*)

Abstract: Combined with best-fit approximation algorithm and genetic algorithm, a new hybrid genetic algorithm is proposed in this paper for bin packing problems with time constraints. Its result shows that the hybrid genetic algorithm based on heuristic method is a good candidate for large-scale bin-packing problems with various constraints.

Key words: hybrid genetic algorithm; bin packing; constraint.

0 引言

经典的装箱问题是把一定数量的物品放入容积相同的一些箱子中，要求每个箱子中物品体积之和不超过箱子容积并且所用的箱子数目最少。在计算机科学和工业领域中，装箱问题有着广泛的应用背景，包括多处理器任务调度、内存管理、资源分配、运输计划等[1]，因此对其求解的研究具有广泛的应用价值。从计算复杂性理论来讲，装箱问题是 NP 完全问题，很难精确求解。目前的近似算法如：下次适应、首次适应、降序下次适应、最佳适应等[2]，在解决复杂约束条件下的装箱问题时，结果与物品的体积数据有较大关系，在极端情况下很不理想。遗传算法作为一种模拟自然进化过程的随机性全局优化概率搜索算法，在求解优化问题中显示了优越的性能。本文针对具有时间约束的装箱问题，提出了一种启发式混合遗传算法，并对其进

收稿日期：2001-12-10

作者简介：陈迎春（1972-），男，海军 91388 部队装备部工程师，硕士研究生。

行了求解。

1 约束装箱问题及其数学模型

设有 n 个物品 u_i, $i=1,2,\cdots n$, 体积分别为 $V(u_i)$, 并分别要求在 $t(u_i)$时间内发送到同一目的地。假设每个箱子的容积均为 C, 装箱时间忽略不计, 一次只能运输一个箱子且运输到目的地所花时间为 T, 即相邻两次运输的时间间隔为 T。确定一种装箱方案使得在规定期限内将这 n 个物品送达目的地且所装箱子数最少。

若设所装箱子数为 m, 将这 m 个箱子按装箱的先后顺序分别编号记 $B_1, B_2, \cdots, B_m$, 记 B_j($j=1,2,\cdots,m$)号箱子中所装物品的下标集为 I_j, 显然此时对于 $j_1\neq j_2$ 有 $I_{j_1}\cap I_{j_2}=\Phi$, 且 $\bigcup_{j=1}^{m} I_j=\{1,2,\cdots,n\}$。这样, 上述约束装箱问题的数学模型为:

$$\min\ m$$
$$\text{s.t.}\begin{cases}\sum_{i\in I_j} V(u_i)\leqslant C & j=1,2,\cdots,m\\ j\cdot T\leqslant\min_{i\in I_j}\{t(u_i)\} & j=1,2,\cdots,m\end{cases}\tag{1}$$

2 基于混合遗传算法的问题求解

2.1 混合遗传算法思想

遗传算法(GA)作为一种有效的全局并行优化搜索工具, 具有简单、通用、鲁棒性强的特点, 在求解优化问题中显示了优越的性能, 但它擅长全局搜索, 而局部搜索能力不足, 且容易出现早熟现象[3]。研究表明, GA 可以用极快的速度达到最优解的 90%左右, 但要达到真正的最优解则要花费相当长的时间。一些对比实验表明, 如果兼顾收敛速度和解的品质两个指标, 简单 GA 方法未必比其它搜索算法更好[4], 解决这一问题的方法可考虑对简单 GA 进行适当改进, 如增加交叉约束算子等。而目前较为活跃的研究领域是考虑 GA 与其它方法的集成, 即混合遗传算法。

梯度法、爬山法、模拟退火法、以及基于知识的启发式算法具有很强的局部搜索能力。如果在 GA 的搜索过程中融合这些优化算法的思想, 构成一种启发式混合遗传算法, 可改善简单 GA 的局部搜索能力, 进一步提高优化质量和搜索效率, 以弥补单一优化方法的某些不足之处。

因此, 在这里, 我们将求解装箱问题的最佳适应法与 GA 相结合, 构成一种启发式混合遗传算法对具有时间约束的装箱问题进行求解。其主要思想是: 在整个装箱过程中, 将箱子按其剩余容积从小到大进行排序; 对某一物品, 它总是装到第一个能装下它的箱子中; 装完一个物品后, 若箱子的剩余容积不小于剩余物品的最小体积, 则将箱子按剩余容积从小到大重新排序, 否则作为装好的箱发送。这样的装箱方案既符合最佳适应近似算法的思想, 同时各个箱子所装物品体积之和也不会超过规定容积。

2.2 染色体的编码方法

编码就是把一个问题的可行解从其解空间转换到遗传算法所能处理的搜索空间的转换方

法, 它是 GA 应用成功与否的关键。在这里, 由于物品的装箱顺序可确定一个装箱方案, 结合 Grefenstette 等求解旅行商问题时提出的巡回路线编码方法[4][5], 可根据物品的装箱顺序来表示染色体编码: 假设一个装箱顺序为(u_{i_1}, u_{i_2}, …, u_{i_n}), 其中 i_1, i_2, …, i_n 为 1, 2, …, n 的一个任意排列, 则编码的第一位数字为 i_1; 第 j ($j=2, 3, \cdots, n$) 位数字为 i_j 减去集合{i_1, i_2, …, i_{j-1}}中小于 i_j 的数字个数。例如, 对于装箱顺序(u_3, u_1, u_4, u_2): 第一位的编码为 3, 第二位的编码为 1, 第三位为 2, 第四位为 1, 即染色体编码为(3, 1, 2, 1)。该编码方法使得任意的基因型个体都能对应于一种具有实际意义的装箱方案, 并能进行经典的交叉和变异操作。

2.3 目标函数和适应度函数

在遗传算法的应用中, 必须对约束条件进行处理。常用的方法是罚函数法, 对违反约束的装箱方案实施惩罚。很显然, 延迟时间越长和延迟物品的体积越大, 所受到的惩罚应该越严厉。由于目标必须使所用的箱子数 m 最小, 这意味着每个箱子装完物品后所剩余的体积尽可能地小。于是我们以减少箱子中物品占有的体积, 从而增大箱子剩余的体积作为惩罚。当处罚因子足够大时, 因违反规定而减少的体积很大, 这就迫使最优装箱方案满足约束条件, 而成为约束问题的最优解。基于此, 我们构造如下罚函数:

$$G_{ij}=\alpha\cdot\max[0, j\cdot T-t(u_i)]\cdot V(u_i) \tag{2}$$

其中, G_{ij} 表示第 j 个箱子中第 i 个物品延迟所受的惩罚, α 为惩罚因子, $\alpha>0$。在试算中, 如果发现最后得出的结果仍违反了约束条件, α 的取值可适当增加, 但惩罚体积不得超过物品本身的体积。

这样, B_j 箱所装物品的体积减去惩罚体积后的剩余体积 F_j 为

$$F_j=\sum_{i\in I_j}\max\{0, V(u_i)-G_{ij}\} \tag{3}$$

这样我们可取以下的目标函数:

$$\max f(X)=\frac{\sum_{j=1}^{m}(F_j/C)^q}{m} \tag{4}$$

其中, C 为箱子容积; q 为常数, 且 $q>1$。该函数既考虑了使所用箱子数最小, 又考虑了使每个箱子装完物品及违反约束受到惩罚后所剩余的容积尽可能地小。

由于各个个体被遗传到下一代群体中的概率是由该个体的适应度来确定的, 适应度越大, 被遗传到下一代的概率就越大。因目标函数为正且求最大值, 所以在应用遗传算法求解时适应度函数可取目标函数。

需说明的是罚函数和目标函数的取法不是唯一的, 但罚函数最好能同时考虑延误时间和延迟物品的体积, 目标函数同时考虑所用箱子数和罚函数。

2.4 交叉、变异和选择算子

交叉是指对两个相互配对的染色体以某种方式相互交换部分基因, 从而形成两个新的个体。最常用和最基本的算子是单点交叉算子, 其具体执行过程如下[3]: (1)群体中的个体进行两两配对。(2)对每一对相互配对的个体, 随机设置某一基因座之后的位置为交叉点。若染色体的长度为 n, 则共有($n-1$)个可能的交叉点位置。(3)对每一对相互配对的个体, 依设定的交叉概率的交叉点相互交换两个个体的部分染色体, 从而产生出两个新的个体。

变异运算是指将个体染色体编码串中的某些基因座上的基因值用该基因座的其他等位基

因来替换,从而形成一个新的个体。常用的均匀变异是用符合某一范围内均匀分布的随机数,以某一较小的概率来替换个体编码串中某个基因座上的原有基因值。

遗传算法使用选择算子来对群体中的个体进行优胜劣汰:适应度较高的个体被遗传到下一代群体中的概率较大;适应度较低的个体被遗传到下一代群体中的概率较小。本文采用的最优保存策略可保证迄今为止所得到的最优个体不会被交叉、变异等操作所破坏。其具体过程是:(1)找出当前群体中适应度最高的个体和适应度最低的个体。(2)若当前群体中最佳个体的适应度比迄今为止的最好个体的适应度还要高,则以当前群体中的最佳个体作为新的迄今为止的最好个体。(3)用迄今为止的最好个体替换当前群体中的最差个体。

2.5 算法步骤

选择算子采用比例选择算子,同时采取最优保存策略;交叉算子采用单点交叉算子;变异算子采用编码字符集 $V=\{B_1, B_2, \cdots, B_m\}$ 范围内的均匀随机变异。混合遗传算法的步骤可描述如下:

(1)进化代数计算器初始化: $g \leftarrow 0$;

(2)随机产生初始群体 $P(g)$;

(3)对 $P(g)$ 用最佳适应法装箱以产生最好的个体;

(4)评价群体 $P(g)$ 的适应度;

(5)终止条件判断。若满足终止条件,输出当前最优个体,算法结束;否则继续以下进化过程:

a. 个体交叉操作: $p'(g) \leftarrow \mathrm{crossover}[p(g)]$;

b. 个体变异操作: $p''(g) \leftarrow \mathrm{mutation}[p'(g)]$;

c. 用 2.1 节中的最佳适应法装箱改进 $p''(g)$

d. 评价 $p''(g)$ 的适应度;

e. 个体选择操作: $p(g+1) \leftarrow \mathrm{selection}[p(g) \cup p''(g)]$

f. $g \leftarrow g+1$, 返回(5)

3 实例分析

设含有 18 个物品的装箱问题:

$$U=\{u_1, u_2, \cdots, u_{18}\}$$

$$V(u_i)=\begin{cases} 0.143 & 1 \leqslant i \leqslant 6 \\ 0.333 & 7 \leqslant i \leqslant 12 \\ 0.500 & 13 \leqslant i \leqslant 18 \end{cases}$$

满足送达时间要求, $t=\{4, 4, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 9, 9, 9, 9, 9, 9\}$。我们假定箱子容积 $C=1$, 装箱时间忽略不计, 每个箱子的运输间隔时间 $T=1$, 即第 1 到第 6 号物品必须装在前 3 个箱子中, 第 7 到第 12 号物品必须装在前 2 个箱子中, 第 13 到第 18 号物品必须装在前 8 个箱子中。

取遗传算法运行参数为{群体大小, 进化代数、交叉概率、变异概率}={100, 50, 0.85, 0.05}, $\alpha=1.0$、$q=2$, 运算得到装箱方案的箱子数为 6, 最大适应度为 0.976, 装箱方案为:

$B_1=\{u_8, u_{11}, u_{12}\}$,　$B_2=\{u_9, u_7, u_{10}\}$,　$B_3=\{u_2, u_6, u_5, u_3, u_1, u_4\}$

$B_4=\{u_{14}, u_{15}\}$,　$B_5=\{u_{17}, u_{16}\}$,　$B_6=\{u_{18}, u_{13}\}$

结果表明, 约束条件下的装箱问题适合用遗传算法求解, 用混合遗传算法能得到与最优方案很近似的结果。

在试算中, 我们发现 α 的取值对结果影响较大: 在群体大小为 100、交叉概率 0.85、变异概率 0.05 的条件下, 当 α 小于 0.3 时, 群体进化 5000 代仍不能收敛到最优值; 当 α=2 时进化 23 代就可得到最优值; α 取值在 1.0 左右时平均进化代数在 50 以下, 结果比较稳定。图 1 为 α 取值与得到最优结果的进化代数关系图。

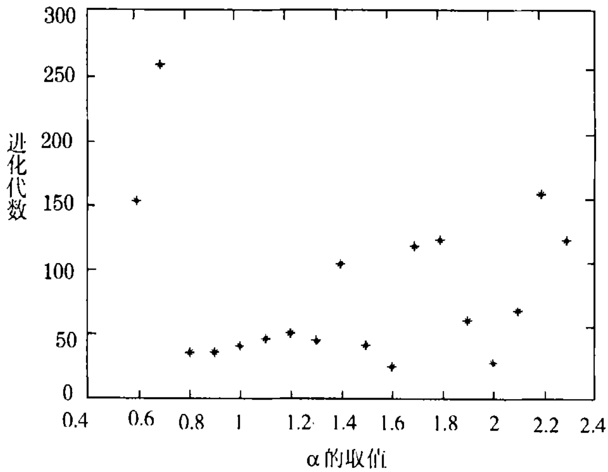

图 1　α 取值与得到最优结果进化代数关系

4　结束语

本文提出的基于启发式近似算法的混合遗传算法亦可推广到复杂约束的装箱问题求解: 箱子目的地不同; 一次可运输多个箱子; 箱子容积不等; 物品价值不等; 某些物品不能混装等。此时在设计罚函数和目标函数时需综合考虑这些约束条件, 但并不影响算法的使用。

参考文献

[1] Johnson D S. Fast algorithms for bin packing[J]. Journal of Computer and System Sciences, 1974, 8(3): 272-314.

[2] Coffman E G, Garey M R, Johnson D S. Approximation algorithms for bin packing: A survey[A]. Hochbaum D ed. Approximation Algorithms for NP-Hard Problems[C]. Boston: PWS Publishing, 1996. 46-93.

[3] 周明, 孙树栋. 遗传算法原理及应用[M]. 国防工业出版社, 1999.

[4] 陈莉. 混合遗传算法及应用[J]. 四川师范大学学报(自然科学版). 1998, 21(5): 553-558.

[5] Grefenstette J, et. Al, Genetic Algorithms for the Traveling Salesman Problem[A]. Proc. of 1st Int. Conf. on Genetic Algorithms and their Applications[C]. Mahwah: Lawrence Erlbaum Associates, 1985. 160-168.

[6] Michalewicz z bigniew. 演化程序——遗传算法和数据编码的结合[M]. 北京: 科学出版社, 2000.